

# Multimodal Hierarchical Reinforcement Learning Policy for Task-Oriented Visual Dialog

**Jiaping Zhang**

University of California, Davis  
jpszhang@ucdavis.edu

**Tiancheng Zhao**

Carnegie Mellon University  
tianchez@cs.cmu.edu

**Zhou Yu**

University of California, Davis  
joyu@ucdavis.edu

## Abstract

Creating an intelligent conversational system that understands vision and language is one of the ultimate goals in Artificial Intelligence (AI) (Winograd, 1972). Extensive research has focused on vision-to-language generation, however, limited research has touched on combining these two modalities in a goal-driven dialog context. We propose a multimodal hierarchical reinforcement learning framework that dynamically integrates vision and language for task-oriented visual dialog. The framework jointly learns the multimodal dialog state representation and the hierarchical dialog policy to improve both dialog task success and efficiency. We also propose a new technique, state adaptation, to integrate context awareness in the dialog state representation. We evaluate the proposed framework and the state adaptation technique in an image guessing game and achieve promising results.

## 1 Introduction

The interplay between vision and language has created a range of interesting applications, including image captioning (Karpathy and Fei-Fei, 2015), visual question generation (VQG) (Mostafazadeh et al., 2016), visual question answering (VQA) (Antol et al., 2015), and reference expressions (Hu et al., 2016). Visual dialog (Das et al., 2017b) extends the VQA problem to multi-turn visual-grounded conversations without specific goals. In this paper, we study the task-oriented visual dialog setting that requires the agent to learn the multimodal representation and dialog policy for decision making. We argue that a task-oriented visual intelligent conversational sys-

tem should not only acquire vision and language understanding but also make appropriate decisions efficiently in a situated environment. Specifically, we designed a 20 images guessing game using the Visual Dialog dataset (Das et al., 2017a). This game is the visual analog of the popular 20 question game. The agent aims to learn a dialog policy that can guess the correct image through question answering using the minimum number of turns.

Previous work on visual dialogs (Das et al., 2017a,b; Chattopadhyay et al., 2017) focused mainly on vision-to-language understanding and generation instead of dialog policy learning. They let an agent ask a fixed number of questions to rank the images or let humans make guesses at the end of the conversations. However, such setting is not realistic in real-world task-oriented applications, because in task-oriented applications, not only completing the task successfully is important but also completing it efficiently. In addition, the agent should also be informed of the wrong guesses, so that it becomes more aware of the vision context. However, solving such real-world setting is a challenge. The system needs to handle the large dynamically updated multimodal state-action space and also leverage the signals in the feedback loop coming from different sub-tasks.

We propose a *multimodal hierarchical reinforcement learning* framework that allows learning visual dialog state tracking and dialog policy jointly to complete visual dialog tasks efficiently. The framework we propose takes inspiration from feudal reinforcement learning (FRL) (Dayan and Hinton, 1993), where levels of hierarchy within an agent communicate via explicit goals in a top-down fashion. In our case, it decomposes the decision into two steps: a first step where a master policy selects between verbal task (information query) and vision task (image retrieval), and a second step where a primitive action (question or im-

age) is chosen from the selected task. Hierarchical RL that relies on space abstraction, such as FRL, is useful to address the challenge of large discrete action space and has been shown to be effective in dialog systems, especially for large domain dialog management (Casanueva et al., 2018). Besides, we propose a new technique called *state adaptation* in order to make the multimodal dialog state more aware of the constantly changing visual context. We demonstrate the efficacy of this technique through ablation analysis.

## 2 Related Work

### 2.1 Visual Dialog

Visual dialog requires the agent to hold a multi-turn conversation about visual content. Several visual dialog tasks have been developed, including image grounded conversation generation (Mostafazadeh et al., 2017). Guess What?! (De Vries et al., 2017) involves locating visual objects using dialogs. VisDial (Das et al., 2017a) situates an answer-bot (A-Bot) to answer questions from a question-bot (Q-Bot) about an image. Das et al. (2017b) applied reinforcement learning (RL) to the VisDial task to learn the policies for the Q/A-Bots to collaboratively rank the correct image among a set of candidates. However, their Q-Bot can only ask questions and cannot make guesses. Chattopadhyay et al. (2017) further evaluated the pre-trained A-bot in a similar setting to answer human generated questions. Since humans are tasked to ask questions, the policy learning of Q-Bot is not investigated. Finally, (Manuvinakurike et al., 2017) proposed an incremental dialogue policy learning method for image guessing. However, their dialog state only used language information and did not include visual information. We build upon prior works and propose a framework that learns an optimal dialog policy for the Q-Bot to perform both question selection and image guessing through exploiting multimodal information.

### 2.2 Reinforcement Learning

RL is a popular approach to learn an optimal dialog policy for task-oriented dialog systems (Singh et al., 2002; Williams and Young, 2007; Georgila and Traum, 2011; Lee and Eskenazi, 2012; Yu et al., 2017). The deep Q-Network (DQN) introduced by Mnih et al. (2015) achieved human-level performance in Atari games based on deep neural networks. Deep RL was then used to jointly

learn the dialog state tracking and policy optimization in an end-to-end manner (Zhao and Eskenazi, 2016). In our framework, we use a DQN to learn the higher level policy for question selection or image guessing. Van Hasselt et al. (2016) proposed a double DQN to overcome the overestimation problem in the Q-Learning and Schaul et al. (2015) suggested prioritized experience replay to improve the data sampling efficiency for training DQN. We apply both techniques in our implementation. One limitation of DQNs is that they cannot handle unbounded action space, which is often the case for natural language interaction. He et al. (2015) proposed Deep Reinforcement Relevance Network (DRRN) that can handle inherently large discrete natural language action space. Specifically, the DRRN takes both the state and natural language actions as inputs and computes a Q-value for each state action pair. Thus, we use a DRRN as our question selection policy to approximate the value function for any question candidate.

Our work is also related to hierarchical reinforcement learning (HRL) which often decomposes the problem into several sub-problems and achieves better learning convergence rate and generalization compared to flat RL (Sutton et al., 1999; Dietterich, 2000). HRL has been applied to dialog management (Lemon et al., 2006; Cuayáhuitl et al., 2010; Budzianowski et al., 2017) which decomposes the dialog policy with respect to system goals or domains. When the system enters a sub-task, the selected dialog policy will be used and continue to operate until the sub-problem is solved, however the terminate condition for a subproblem has to be predefined. Different from prior work, our proposed architecture uses hierarchical dialog policy to combine two RL architectures within a control flow, i.e., DQN and DRRN, in order to jointly learn multimodal dialog state representation and dialog policy. Note that our HRL framework resembles the FRL hierarchy (Dayan and Hinton, 1993) that exploits space abstraction, state sharing and sequential execution.

## 3 Proposed Framework

Figure 2 shows an overview of the multimodal hierarchical reinforcement learning framework and the simulated environment. There are four main modules in the framework. The **visual dialog semantic embedding** module learns a multimodal dialog state representation to support the **visual**

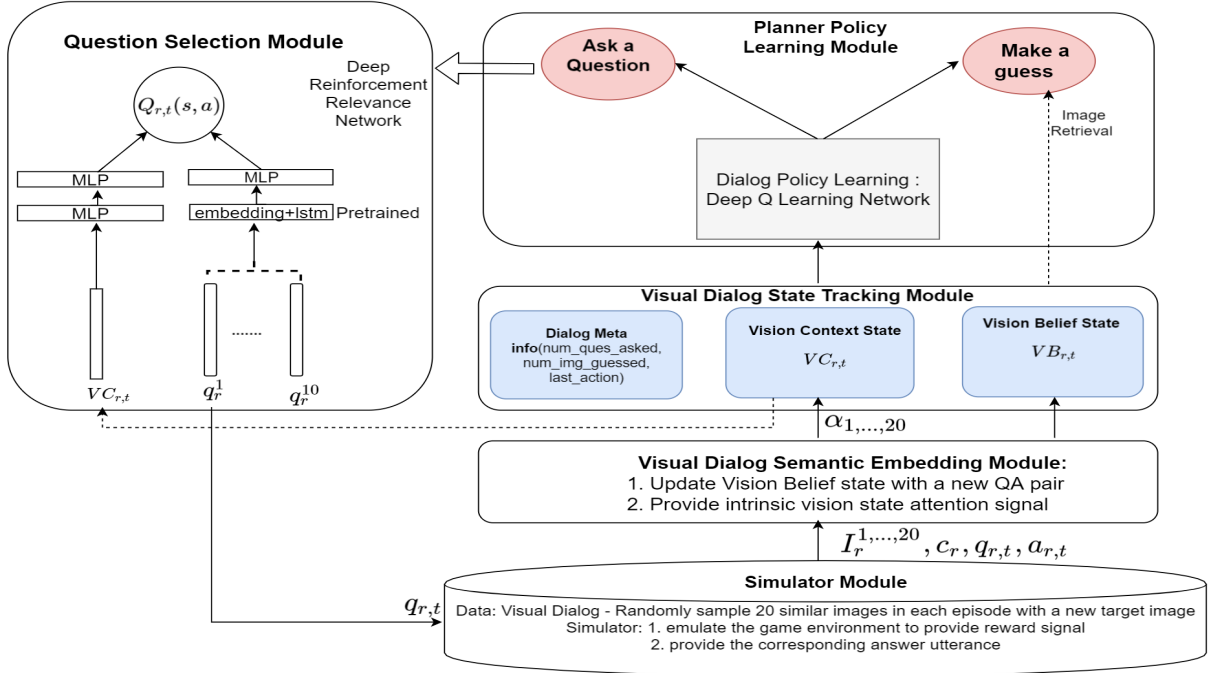


Figure 1: The information flow of the multimodal hierarchical reinforcement learning framework

**dialog state tracking** module with attention signals. Then the **hierarchical policy learning** module takes the visual dialog state as the input to optimize the high-level control policy between **question selection** and image retrieval.

### 3.1 Visual Dialog Semantic Embedding

This module learns the multimodal representation for the downstream visual dialog state tracking. Figure 3 shows the network architecture for pre-training the visual dialog semantic embedding. A VGG-19 CNN (Simonyan and Zisserman, 2014) and a multilayer perceptron (MLP) with L2 normalization are used to encode visual information (images) as a vector  $I \in R^k$ . We use a dialog-conditioned attentive encoder (Lu et al., 2017) to encode textual information as a vector  $T \in R^k$  where  $k$  is the joint embedding size. The image caption ( $c$ ) is encoded with a LSTM to get a vector  $m^c$  and each QA pair ( $H_0, \dots, H_t$ ) is encoded separately with another LSTM as  $M_t^h \in R^{d \times t}$  where  $t$  is the turn index and  $d$  is the LSTM embedding size. Conditioned on the image caption embedding, the model attends to the dialog history:

$$z_t^h = w_a^T \tanh(W_h M_t^h + (W_c m^c) \mathbb{1}^T) \quad (1)$$

$$\alpha_t^h = \text{softmax}(z_t^h) \quad (2)$$

where  $\mathbb{1}$  is a vector with all elements set to 1,  $W_h, W_c \in R^{t \times d}$  and  $w_a \in R^k$  are parameters to be learned.  $\alpha \in R^k$  is the attention weight over history. The attended history feature  $\hat{m}_t^h$  is the weighted sum of each column of  $M_t^h$  with  $\alpha_t^h$ . Then  $\hat{m}_t^h$  is concatenated with  $m^c$  and encoded via MLP and  $l_2$  norm to get the final textual embedding ( $T$ ). We train the network with pairwise ranking loss (Kiros et al., 2014) on cosine similarities between the textual and visual embedding. The pretraining step allows the module to have better generalization and improve convergence performance in the RL training.

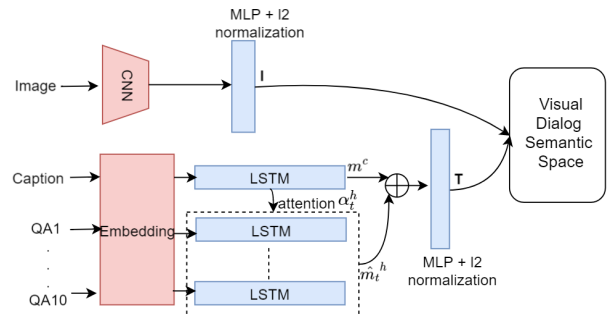


Figure 2: Pretraining scheme of the visual dialog semantic embedding module

Given the QA pairs from the simulated environ-

ment, the output of this module can also be used for the image retrieval sub-task. To verify the quality of this module, we perform a sanity check on an image retrieval task, similar to (Das et al., 2017b). We used the output of the module to rank the 20 images in the game setting. Among 1000 games, we achieved 96.8% accuracy for recall@1 (the target image ranked the highest), which means that this embedding module can provide reliable reward signal in an image retrieval task for the RL training if given the relevant dialog history.

### 3.2 Visual Dialog State Tracking

This module utilizes the output from the visual dialog semantic embedding to formulate the final dialog state representation. We track three types of state information, the dialog meta information (*META*), the vision belief (*VB*) and the vision context (*VC*). The dialog meta information includes the number of questions asked, the number of images guessed and the last action. The vision belief state is the output of the visual dialog semantic embedding module, which captures the internal multimodal information of the agent. We initialize the VB with only the encoding of the image caption and update it with each new incoming QA pair. The vision context state represents the visual information of the environment. In order to make the agent more aware of the dynamic visual context and which images to attend more, we introduce a new technique called *state adaptation* as it updates the vision context state with the attention scores. The *VC* is initialized as the average of image vectors and updated as follows:

$$\alpha_{r,t,i} = \text{sigmoid}(\text{VB}_{r,t} \cdot I_{r,i}) \quad (3)$$

$$\text{VC}_{r,t} = \frac{\sum_{i=1}^{20} \alpha_{r,t,i} I_{r,i}}{\sum_{i=1}^{20} \alpha_i} \quad (4)$$

where  $r, t$  and  $i$  refer to episode, dialog turn and image index. The *VC* is then adjusted based on the attention scores (see equation 4). The attention scores calculated by dot product in the equation 3 represent the affinity between the current vision belief state and each image vector. In the case of wrong guesses (informed by the simulator), we set the attention score for that wrong image to zero. This method is inspired by Tian et al. (2017) who explicitly weights context vectors by context-query relevance for encoding dialog context. The question selection sub-task also takes the

vision context state as input and the vision belief state is used in the image retrieval sub-task.

### 3.3 Hierarchical Policy Learning

The goal is to learn a dialog policy that makes decisions based on the current visual dialog state, i.e, asking a question about the image or making a guess about the image that the user is thinking of. As the agent is situated in a dynamically changing vision context to update its internal decision-making model (approximated by the belief state) with new dialog exchange, we treat such environment as a Partially Observable Markov Decision Process (POMDP) and solve it using deep reinforcement learning. We now describe the key components:

**Dialog State** comes from the visual dialog state tracking module as mentioned in Section 3.2

**Policy Learning:** Given the above dialog state, we introduce a hierarchical dialog policy that contains a high-level control policy and a low-level question selection policy. We learn the control policy with a Double DQN that decides between “question” or “guess” at a game step.

If the high-level action is a “question”, then the control is passed over to the low-level policy, which needs to select a question. One challenge is that the list of candidate questions are different for every game, and the number of candidate questions for different images is also different as well. This prohibits us using a standard DQN with fixed number of actions. He et al. (2015) showed that modeling state embedding and action embedding separately in DRRN has superior performance than per-action DQN as well as other DQN variants for dealing with natural language action spaces. Therefore, we use the DRRN to solve this problem, which computes a matching score between the shared current vision context state and the embedding of each question candidate. We use a softmax selection strategy as the exploration policy during the learning stage. The hierarchical policy learning algorithm is described in the Appendix *Algorithm 1*.

If the high-level action is “guess”, then an image is retrieved using cosine distance between each image vector and the vision belief vector. It is worth mentioning that although the action space of the image retrieval sub-task can be incorporated into a flat DRRN combined with text-based inputs, the training is unstable and does not converge



within this flat RL framework. We suspect this is due to the sample efficiency problem with large multimodal action space for which the question action or guess action typically results in different reward signals. Therefore, we did not compare our proposed method against a flat RL model.

**Rewards:** The reward function is decomposed as  $R = R_G + R_Q + R_I$  where  $R_G$  means the final game reward (win/loss =  $\pm 10$ ),  $R_I$  refers to wrong guess penalty (-3). We define  $R_Q$  as the pseudo reward for the sub-task of question selection as

$$R_Q = A_t - A_{t-1} \quad (5)$$

$$A_t = \text{sigmoid}(\text{VB}_{r,t} \cdot I_{\text{target}}) \quad (6)$$

where  $t$  refers to the dialog turn and *affinity scores* ( $A_t$  and  $A_{t-1}$ ) are the outputs of the sigmoid function that scales the similarity score (0-1) of the vision belief state and the target image vector. The intuition is that different questions provide various information gains for the agent. The integration of  $R_Q$  is a *reward shaping* (Ng et al., 1999) technique that aims to provide immediate rewards to make the RL training more efficient. At each turn, if the verbal task (question selection) is chosen, the  $R_Q$  would serve as immediate reward for training the DQN and DRRN while if the vision task (image retrieval) is chosen, only the  $R_I$  is available for training DQN. At the end of a game, the reward function varies based on the primitive action and the final game result.

### 3.4 Question Selection

The question selection module selects the best question in order to acquire relevant information to update the image belief state. As discussed in Section 3.3, we used a discriminative approach to select the next question for the agent by learning the policy in a *DRRN*. It leverages the existing question candidate pool that is constructed differently with respect to different experiment settings in Section 4.4. Ideally we would like to generate realistic questions online towards a specific goal (Zhang et al., 2017) and we leave this generative approach for future study.

## 4 Experiments

We first describe the simulation of the environment. Then, we talk about different dialog policy models and implementation details. Finally, we discuss three different experimental settings to evaluate the proposed framework.

### 4.1 Simulator Construction

We constructed a simulator for 20 images guessing game using the *VisDial* dataset. Each image corresponds to a dialog consisting of ten rounds of question answering generated by humans. To make the task setting meaningful and the training time manageable, we pre-process and select 1000 sets of games consisting of 20 similar images. The simulator provides the reward signals and answers related to the target image. It also tracks the internal game state. A game is terminated when one of the three conditions is fulfilled: 1) the agent guesses the correct answer, 2) the max number of guesses is reached (three guesses) or 3) the max number of dialog turns is reached. The agent wins the game when it guesses the correct image. If the agent wins the game, it gets a reward of 10, and if the agent loses the game, it gets a reward of -10. The agent also receives a -3 penalty for each wrong guess.

### 4.2 Policy Models

To evaluate the contribution of each technique in the multimodal hierarchical framework: the hierarchical policy, the state adaptation, and the reward shaping, we evaluate five different policy models and perform ablation analysis. We describe each model as follows:

- *Random Policy (Rnd)*: The agent randomly selects a question or makes a guess at any step.
- *Random Question+DQN (Rnd+DQN)*: The agent randomly selects a question but a DQN is used to optimize the hierarchical decision of making a guess or asking a question.
- *DRRN+DQN (HRL)*: Similar to Rnd+ DQN, except that a DRRN is used to optimize the question selection process
- *DRRN+DQN+State Apdation (HRL+SA)*: Similar to HRL, except incorporating the state adaptation, which is similar to the attention re-weighting concept in the vision context state.
- *DRRN+DQN+State Apdation+Reward Shaping (HRL+SAR)*: Similar to HRL+SA, except that reward shaping is applied.

### 4.3 Implementation Details

The details about data pre-processing and training hyper-parameters are described in the Appendix. During the training, the DQN uses the  $\epsilon$ -greedy policy and the DRRN uses the softmax policy for exploration, where  $\epsilon$  is linearly decreased from 1

to 0.1. The resulting framework was trained up to 20,000 iterations for Experiment 1 and 95,000 iterations for Experiment 2 and 3, and evaluated at every 1000 iterations with greedy policy. At each evaluation we record the performance of different models with a greedy policy for 100 independent games. The evaluation metrics are the *win rate* and the *average number of dialog turns*.

#### 4.4 Experimental Setting

We conduct three sets of experiments to explore the effectiveness of the proposed multimodal hierarchical reinforcement learning framework in a real-world scenario step by step. The first experiment constrains the agent to select among the 10 human generated question-answer pairs. This setting enables us to assess the effectiveness of the framework in a less error-prone setting. The second experiment does not require a human to generate the answer to emulate a more realistic environment. Specifically, we enlarge the number of questions by including 200 human generated questions for the 20 images, and use a pre-trained visual question answer model to generate answers with respect to the target image. In the last experiment, we further automate the process by generating questions given the 20 images using a pre-trained visual question generation model. So the agent does not require any human input with respect to any image for training.

### 5 Results

We evaluate the models described in Section 4.2 under the settings described in Section 4.4 and report results as following.

#### 5.1 Experiment 1: Human Generated Question-Answer Pairs

The agent selects the next question among the 10 question-answer pairs human generated and want to identify the targeted image accurately and efficiently through natural language conversation. We terminate the dialog after ten turns. Each model’s performance is shown in Table 1. *HRL+SAR* achieves the best win rate with statistical significance. The *HRL+SAR* policy model performs much better than methods without hierarchical control structure and state adaptation. The learning curves in Figure 4 and 5 reveal that the *HRL+SAR* converges faster. We further perform bootstrap tests by resampling the game results

from each experiment with replacement 1,000 times. Then we calculate the probability of significance level for the difference of average win rates or average turn length to check whether the relative performance improvement from the last baseline is statistically significant. The result shows that the *question selection (DRRN)* and *state adaptation* bring the most significant performance improvements ( $p < 0.01$ ) while *reward shaping* has less impact ( $p < 0.05$ ). We also observe that the average number of turns with hierarchical policy learning (*HRL*) is slightly longer than that of *Rnd+DQN* but with less statistically significant difference. This is probably because this setting provides the 10 predefined question-answer pairs with a smaller action space, the *DQN* model tends to encourage the agent to make guesses quicker, while policy models with hierarchical structures tends to optimize the overall task completion rate.

	Win Rate(%)	Avg Turn
Random Policy	28.3	5.13
Random Question + DQN	42.7 ***	6.68 ***
DRRN + DQN	51.5 ***	6.97 *
DRRN + DQN + State adaptation	71.3 ***	7.12
DRRN + DQN + State adaptation + Reward Shaping	<b>76.3 **</b>	7.22

\*\*\*( $p < 0.01$ ), \*\*( $p < 0.05$ ) and \*( $p < 0.1$ )

Table 1: Model Performance in Experiment 1

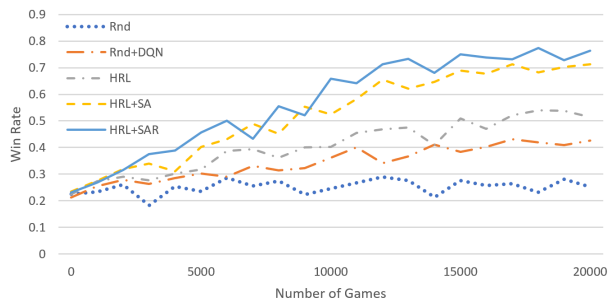


Figure 3: Learning curves of win rates for five different policy policies in Experiment 1

We find that RL methods (DQN & DRRN) significantly improve the win rate as they learn to select the optimal list of questions to ask. We also observe that our proposed state adaptation method

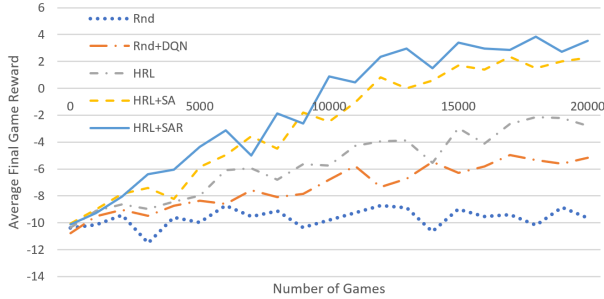


Figure 4: Learning curves of final rewards for five different dialog policies in Experiment 1

for vision context state helps achieve the largest performance improvement. The hierarchical control architecture and the state abstraction sharing (Dietterich, 2000) also improve both learning speed and agent performance. This aligns with the observation in Budzianowski et al. (2017).

Moreover, on average, we observe that after seven turns, the agent was able to select the target image with a sufficiently high success rate. We further explore if the proposed hierarchical framework enables efficient decision-making when compared to the agent that keeps asking questions and only makes the guess at the end of the dialog. We refer to such models as the oracle baselines. For example, the Oracle@7 makes the guess at the 7th turn based on the previous dialog history with the correct order of question-answer pairs in the dataset. The oracle baselines are strong, since they represent the best performance the model can get given the optimal question order provided by human.

	number of rounds	win rate(%)
Oracle Baselines	7	<b>69.7</b>
	8	<b>77.5</b>
	9	87.8
	10	92.4

Table 2: Oracle baselines Performance

Table 2 shows the performance of the oracle baselines with various fixed turns. We performed significance tests between each oracle baseline and the hierarchical framework. Since our hierarchical framework requires on average 7.22 turns to complete, so we compared it with Oracle@7 and Oracle@8. We found that the proposed method outperforms Oracle@7 with  $p$ -value  $< 0.01$ , and similar to Oracle@8 (significant difference

( $p$ -value  $> 0.1$ ). The reason that the hierarchical framework can outperform Oracle@7 is that it learns to make a guess whenever the agent is confident enough, therefore achieving better win rate. Oracle@8 in general receives more information as the dialogs are longer, therefore has an advantage over the hierarchical method. However, it still performs similar to the proposed method, which demonstrates that by learning the hierarchical decision, it enables the agent to achieve the goal more efficiently. One thing we need to point out is that the proposed method also received extra information about whether the guess is correct or not from the environment. Oracle baselines do not have such information, as it can only make a guess at the end of the dialog. Oracle@9 and @10 are better than the hierarchical framework statistically, because they acquire much more information by having longer turns.

## 5.2 Experiment 2: Questions Generated by Human and Answers Generated Automatically

To make the experimental setting more realistic, we select 200 questions generated by a human with respect to 20 images provided and create a user simulator that generates the answers related to the target image. Here, as the questions space is larger, we terminate the dialog after 20 turns. We follow the supervised training scheme discussed in (Das et al., 2017b) to train the visual question generation module offline.

	Win Rate(%)	Avg Turn
Random Policy	15.6	5.67
Random Question + DQN	34.8 ***	18.81 ***
DRRN + DQN	48.7 ***	18.78
DRRN + DQN + State adaptation	62.4 ***	16.93 **
DRRN + DQN + State adaptation + Reward Shaping	<b>67.3 **</b>	<b>16.68</b>

\*\*\*( $p < 0.01$ ), \*\*( $p < 0.05$ ) and \*( $p < 0.1$ )

Table 3: Model Performance in Experiment 2

Results in Table 3 indicate that *HRL+SAR* significantly outperforms *Rnd* and *Rnd+DQN* in both win rate and average number of dialog turns. The setting in Experiment 2 is more challenging than that of Experiment 1, because the visual ques-

tion module introduces noise that can influence the policy learning. However, the noise also simulates the real-world scenario that a user might have an implicit goal that may change within the task. A user can also accidentally make errors in answering the question. The proposed hierarchical framework (*HRL+SAR*) with state adaptation and reward shaping achieves the best win rate and the least number of dialog turns in this noisy experiment setting. As compared to Experiment 1, the policy models with hierarchical structures can both optimize the overall task completion rate and the dialog turns. We did not report oracle baselines results, since the oracle order of all the questions (ideally generated by humans) was not available.

### 5.3 Experiment 3: Question-Answer Pairs Generated Automatically

In this setting, both questions and answers are generated automatically through pre-trained visual question and answer generation models (Das et al., 2017b). Such setting enables the agent to play the guessing game given *any* image as no human input of the image is needed. Notice that the answers should be generated with respect to a target image for our task setting. In this setting, we also set the maximum number of dialog turns to be 20.

	Win Rate(%)	Avg Turn
Random Policy	12.4	5.79
Random Question + DQN	18.4 **	19.43 ***
DRRN + DQN	35.6 ***	19.33
DRRN + DQN + State adaptation	44.8 **	18.84 *
DRRN + DQN + State adaptation + Reward Shaping	<b>48.3 **</b>	<b>18.77</b>

\*\*\*( $p < 0.01$ ), \*\*( $p < 0.05$ ) and \*( $p < 0.1$ )

Table 4: Model Performance in Experiment 3

The results in Table 4 show that the performance of the three policies significantly dropped compared to Experiment 2. Such observation is expected, as the noise coming from both the visual question and answer generation module increases the task difficulty. However, the proposed *HRL+SAR* is still more resilient to the noise and achieves a higher win rate and less average number of turns compared to other baselines. Figure 5 from the Appendix shows that in Experiment 2

the agent tends select relevant questions faster to ask although the answers can be misleading. On the other hand, in Experiment 3, the agent reacts to the generated question and answers slower to complete the task. The model performance decreases when we increase the task difficulty in order to emulate the real-world scenarios. It hints that there is a possible limitation of using the *Vis-Dial* dataset, because the dialog is constructed by users who casually talk about MS COCO images (Chen et al., 2015) instead of exchanging with an explicit contextual goal in the dialog.

## 6 Discussion and Future Work

We develop a framework for task-oriented visual dialog systems and demonstrate the efficacy of integrating multimodal state representation with hierarchical decision learning in an image guessing game. We also introduce a new technique called state adaptation to further improve the task performance through integrating context awareness. We also test the proposed framework in various noisy settings to simulate real-world scenarios and achieve robust results.

The proposed framework is practical and extensible for real-world applications. For example, the designed system can act as a fashion shopping assistant to help customers pick clothes through strategically inquiring their preferences while leveraging vision intelligence. In another application, such as criminology practice, the agent can communicate with witnesses to identify suspects from a large face database.

Although games provide a rich domain for multimodal learning research, admittedly it is challenging to evaluate a multimodal dialog system due to the data scarcity problem. In future work, we would like to extend and apply the proposed framework for human studies in a situated real-world application, such as a shopping scenario. We also plan to incorporate domain knowledge and database interactions into the system framework design, which will make the dialog system more flexible and effective. Another possible extension of the framework is to update the off-line question and answer generation modules with an online generative version and retrain the module with reinforcement learning.



## References

- Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433.
- Pawel Budzianowski, Stefan Ultes, Pei-Hao Su, Nikola Mrksic, Tsung-Hsien Wen, Inigo Casanueva, Lina Rojas-Barahona, and Milica Gasic. 2017. Subdomain modelling for dialogue management with hierarchical reinforcement learning. *arXiv preprint arXiv:1706.06210*.
- Inigo Casanueva, Paweł Budzianowski, Pei-Hao Su, Stefan Ultes, Lina Rojas-Barahona, Bo-Hsiang Tseng, and Milica Gašić. 2018. Feudal reinforcement learning for dialogue management in large domains. *arXiv preprint arXiv:1803.03232*.
- Prithvijit Chattopadhyay, Deshraj Yadav, Viraj Prabhu, Arjun Chandrasekaran, Abhishek Das, Stefan Lee, Dhruv Batra, and Devi Parikh. 2017. Evaluating visual conversational agents via cooperative human-ai games. *arXiv preprint arXiv:1708.05122*.
- Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.
- Heriberto Cuayáhuitl, Steve Renals, Oliver Lemon, and Hiroshi Shimodaira. 2010. Evaluation of a hierarchical reinforcement learning spoken dialogue system. *Computer Speech & Language*, 24(2):395–429.
- Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017a. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2.
- Abhishek Das, Satwik Kottur, José MF Moura, Stefan Lee, and Dhruv Batra. 2017b. Learning cooperative visual dialog agents with deep reinforcement learning. *arXiv preprint arXiv:1703.06585*.
- Peter Dayan and Geoffrey E Hinton. 1993. Feudal reinforcement learning. In *Advances in neural information processing systems*, pages 271–278.
- Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. 2017. Guesswhat?! visual object discovery through multi-modal dialogue. In *Proc. of CVPR*.
- Thomas G Dietterich. 2000. Hierarchical reinforcement learning with the maxq value function decomposition. *J. Artif. Intell. Res.(JAIR)*, 13(1):227–303.
- Kallirroi Georgila and David Traum. 2011. Reinforcement learning of argumentation dialogue policies in negotiation. In *Twelfth Annual Conference of the International Speech Communication Association*.
- Ji He, Jianshu Chen, Xiaodong He, Jianfeng Gao, Li-hong Li, Li Deng, and Mari Ostendorf. 2015. Deep reinforcement learning with a natural language action space. *arXiv preprint arXiv:1511.04636*.
- Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. 2016. Natural language object retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 4555–4564. IEEE.
- Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.
- Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.
- Sungjin Lee and Maxine Eskenazi. 2012. Pomdp-based let’s go system for spoken dialog challenge. In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 61–66. IEEE.
- Oliver Lemon, Xingkun Liu, Daniel Shapiro, and Carl Tollerander. 2006. Hierarchical reinforcement learning of dialogue policies in a development environment for dialogue systems: Reall-dude. In *BRANDIAL06, Proceedings of the 10th Workshop on the Semantics and Pragmatics of Dialogue*, pages 185–186.
- Jiasen Lu, Anitha Kannan, Jianwei Yang, Devi Parikh, and Dhruv Batra. 2017. Best of both worlds: Transferring knowledge from discriminative learning to a generative visual dialog model. In *Advances in Neural Information Processing Systems*, pages 313–323.
- Ramesh Manuvinaurike, David DeVault, and Kallirroi Georgila. 2017. Using reinforcement learning to model incrementality in a fast-paced dialogue game. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 331–341.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529.
- Nasrin Mostafazadeh, Chris Brockett, Bill Dolan, Michel Galley, Jianfeng Gao, Georgios P Spithourakis, and Lucy Vanderwende. 2017. Image-grounded conversations: Multimodal context for natural question and response generation. *arXiv preprint arXiv:1701.08251*.
- Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016. Generating natural questions about an image. *arXiv preprint arXiv:1603.06059*.

Andrew Y Ng, Daishi Harada, and Stuart Russell. 1999. Policy invariance under reward transformations: Theory and application to reward shaping. In *ICML*, volume 99, pages 278–287.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. 2015. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Satinder Singh, Diane Litman, Michael Kearns, and Marilyn Walker. 2002. Optimizing dialogue management with reinforcement learning: Experiments with the njfun system. *Journal of Artificial Intelligence Research*, 16:105–133.

Richard S Sutton, Doina Precup, and Satinder Singh. 1999. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211.

Zhiliang Tian, Rui Yan, Lili Mou, Yiping Song, Yansong Feng, and Dongyan Zhao. 2017. How to make context more useful? an empirical study on context-aware neural conversational models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 231–236.

Hado Van Hasselt, Arthur Guez, and David Silver. 2016. Deep reinforcement learning with double q-learning. In *AAAI*, volume 16, pages 2094–2100.

Jason D Williams and Steve Young. 2007. Partially observable markov decision processes for spoken dialog systems. *Computer Speech & Language*, 21(2):393–422.

Terry Winograd. 1972. Understanding natural language. *Cognitive psychology*, 3(1):1–191.

Zhou Yu, Alan W Black, and Alexander I Rudnicky. 2017. Learning conversational systems that interleave task and non-task content. *IJCAI*.

Junjie Zhang, Qi Wu, Chunhua Shen, Jian Zhang, Jianfeng Lu, and Anton van den Hengel. 2017. Asking the difficult questions: Goal-oriented visual question generation via intermediate rewards. *arXiv preprint arXiv:1711.07614*.

Tiancheng Zhao and Maxine Eskenazi. 2016. Towards end-to-end learning for dialog state tracking and management using deep reinforcement learning. *arXiv preprint arXiv:1606.02560*.

## A Data Pre-Processing and Training Details

After data pre-processing, we had a vocabulary size of 8,957 and image vector dimension of

4,096. To pre-train the visual dialog semantic embedding, we used the following parameters: the size of word embedding is 300; the size of LSTMs is 512; 0.2 dropout rate and the final embedding size 1024 with MLP and l2 norm. We fixed the visual dialog semantic embedding during the RL training. The high-level policy learning module - Double DQN was trained with the following hyperparameters: three MLP layers of sizes 1000, 500 and 50 with tanh activation respectively. For hyper-parameters of DQN, the behavior network was updated every 5 steps and the interval for updating the target network is 500.  $\epsilon$ -greedy exploration was used for training, where  $\epsilon$  is linearly decreased from 1 to 0.1. The question selection module - DRRN encodes the context vector and question vector separately with two MLP layers of sizes 256 and 128 and dot product was used as the interaction function. The experience replay buffer sizes are 25,000 for DQN and 50,000 for DRRN. Both RL networks were trained through RMSProp with batch size 64. Bootstrapping and prioritized replay were also used to facilitate RL training. The reward discount factor was set to be 0.99.

## B Sample Dialog

See Figure 5.

## C Hierarchical Policy Learning Algorithm

See Algorithm 1.

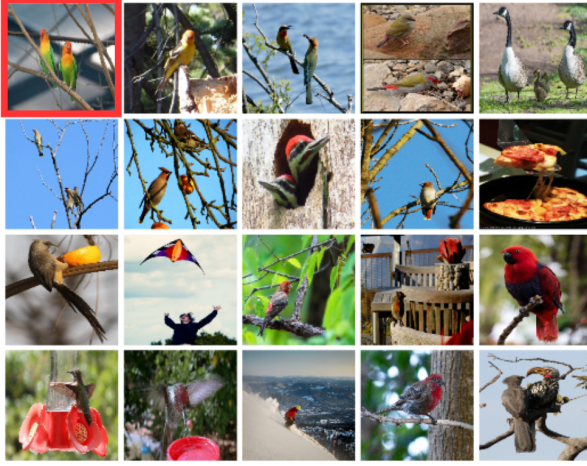
				200 Predefined Questions	200 Generated Questions
				<b>Caption: 2 colorful parrots perched on a tree branch</b>	
				<b>Agent:</b> Are there leaves on the tree ?	<b>Agent:</b> Can you see any people?
				<b>User:</b> No	<b>User:</b> No
				<b>Agent:</b> What color are the birds?	<b>Agent:</b> Is this in the tree ?
				<b>User:</b> red ( <i>wrong answer</i> )	<b>User:</b> No
				<b>Agent:</b> Both are the same colors ?	<b>Agent:</b> Is it daytime ?
				<b>User:</b> Yes	<b>User:</b> Yes
				<b>Agent:</b> Guess [third image on the second row]( <i>wrong guess</i> )	<b>Agent:</b> What color is the sky ?
				<b>Agent:</b> Do they have any yellow colored feathers?	<b>User:</b> I don't know
				<b>User:</b> No ( <i>wrong answer</i> )	<b>Agent:</b> any other animals ?
				<b>Agent:</b> Do they have any green colored feathers?	<b>User:</b> Yes
				<b>User:</b> Yes	<b>Agent:</b> can you see any other birds ?
				<b>Agent:</b> Can you see any other birds ?	<b>User:</b> no
				<b>User:</b> No	<b>Agent:</b> Do they have any green colored feathers ?
				<b>Agent:</b> Guess [First image on the first row](Correct)	<b>User:</b> I can't tell
					<b>Agent:</b> what kind of birds ?
					<b>User:</b> I can't tell
					....

Figure 5: A successful dialog from Experiment 2 and a failure dialog from Experiment 3

---

### Algorithm 1 Hierarchical Policy Learning

---

- 1: Initialize Double DQN(online network parameters  $\theta$  and target network parameters  $\theta^-$ ) and DRRN(network parameters  $\theta^+$ ) with small random weights and corresponding replay memory  $E_{DQN}$  and  $E_{DRRN}$  to capacity  $N$ .
  - 2: Initialize game simulator and load dictionary.
  - 3: **for** episode  $r = 1, \dots, M$  **do**
  - 4:   Restart game simulator.
  - 5:   Receive image caption and candidate images from the simulator, and convert them to representation via pre-trained visual dialog semantic embedding layer, denoted as initial state  $S_{r,0}$
  - 6:   **for**  $t = 1, \dots, T$  **do**
  - 7:     sample high-level action from DQN,  $A_t \sim \pi_{DQN}(S_{r,t})$
  - 8:     **if**  $A_{r,t} = Q$  (asking a question) **then**
  - 9:       Compute  $Q(VC_t, q^i)$  for the list of questions  $Q_{r,t}$  using DRRN forward activation and select the question  $q_{r,t}$  with the max  $Q$ -value, and keep track the next available question pool  $Q_{r,t+1}$
  - 10:     **if**  $A_{r,t} = G$  (guessing an image) **then**
  - 11:       Select the image  $g_{r,t}$  with the smallest cosine distance between an image vector  $I^i$  and current image belief state  $VB_{r,t}$
  - 12:       Execute action  $q_{r,t}$  or  $g_{r,t}$  in the simulator and get the next visual dialog state representation  $S_{r,t+1}$  and reward signal  $R_{r,t}$
  - 13:       Store the transition  $(S_{r,t}, A_{r,t}, S_{r,t+1}, R_{r,t})$  into  $E_{DQN}$  and if asking a question, also store the transition  $(VC_{r,t}, q_{r,t}, VC_{r,t+1}, R_{r,t}, Q_{r,t+1})$  into  $E_{DRRN}$
  - 14:       Sample random mini-batch of transitions  $(S_k, A_k, S_{k+1}, R_k)$  from  $E_{DQN}$
  - 15:       Set  $y_{DQN} = \begin{cases} R_k & \text{if terminal state} \\ R_k + \gamma Q_{DQN}(S_{k+1}, \text{argmax}_{a'} Q(S_{k+1}, a'; \theta); \theta^-) & \text{if else} \end{cases}$
  - 16:       Sample random mini-batch of transitions  $(VC_l, q_l, VC_{l+1}, R_l, Q_{l+1})$  from  $E_{DRRN}$
  - 17:       Set  $y_{DRRN} = \begin{cases} R_l & \text{if terminal state} \\ R_l + \gamma \max_{a' \in Q_{l+1}} Q_{DRRN}(VC_{l+1}, a'; \theta^+) & \text{if else} \end{cases}$
  - 18:       Perform gradient steps for DQN with loss  $\|y_{DQN} - Q_{DQN}(S_k, A_k; \theta)\|^2$  with respect to  $\theta$  and DRRN with loss  $\|y_{DRRN} - Q_{DRRN}(VC_l, q_l; \theta^+)\|^2$  with respect to  $\theta^+$
  - 19:       Replace target parameters  $\theta^- \leftarrow \theta$  for every  $N$  steps.
  - end for**
  - end for**
-